Software Engineering Institute

Army ASSIP System-of-Systems Test Metrics Task

Carol A. Sledge, PhD

November 2006

SPECIAL REPORT
CMU/SEI-2006-SR-011

Software Engineering Institute Army Strategic Software Improvement Program (ASSIP) Task Team

Carnegie Mellon

Table of Contents

List of Figures

Executive Summary

The Army Strategic Software Improvement Program (ASSIP) is a long-term strategic partnership with the Assistant Secretary of the Army (Acquisition, Logistics, and Technology) (ASA(ALT), Program Executive Officers (PEOs), direct reporting Program Managers (PMs), Army Materiel Command Software Engineering Centers (SECs), and the Carnegie Mellon® Software Engineering Institute[1] (SEI). The ASSIP goal is to dramatically improve the acquisition of software-intensive systems by focusing on acquisition programs, people, and production/sustainment and by institutionalizing continuous improvement.

This special report contains information related to one subtask of this effort, conducted during FY06. Many challenges are associated with system-of-systems integration and testing. As a subject matter expert and neutral party, the SEI was engaged to

* explore the current processes and test results/metrics that are used to address system-of-systems integration and testing

* develop findings and recommendations for improvement based on this initial exploration

* recommend future work to further improve the Army's system-of-systems integration and test practices

In support of uncovering necessary information and background, the SEI interviewed key stakeholders and contributors to better understand

* what was there in the timeframe under review (beginning in April 2004) and at the time of the review (April-June, 2006)

* the challenges faced

* the solutions used (to the date of the review)

The Army is in the lead in addressing the many challenges associated with system-of-systems integration and testing, paving the way for the rest of the U.S Department of Defense (DoD). As a result, the information contained in this report is useful to other organizations facing similar challenges.

The SEI conducted its review with respect to a specific set of circumstances at a substantially later point in time (with respect to those circumstances): Our objective was to

* learn from what had occurred

®   Carnegie Mellon is registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

[1]   The Software Engineering Institute is a federally funded research and development center, funded by the Department of Defense and operated by Carnegie Mellon University.

- determine the reasons behind the decisions/events

- discover improvements made and lessons learned by the various stakeholders and organizations

Consequently, we prepared our briefing to

- use that information and knowledge to address the team's specific task

- provide recommendations going forward

As a result of this work, we found the challenges associated with system-of-systems integration and testing

- reach back to events and decisions (much) earlier in the acquisition, development and test life-cycles

- cross multiple organizations, numerous times

- are aggravated by the rapidity of technological change as well as the closer integration of existing assets and systems necessitated by the changing demands of the operational environment

- are magnified by all of the attendant elements and changes resulting from the transition from a system focus/basis to a system-of-systems focus

While specific organizations are mentioned in the recommendations, the recommendations are stated "looking forward, with a system-of-systems viewpoint" and should not be construed as a criticism of any organization.

Abstract

The Army Strategic Software Improvement Program goal is to dramatically improve the acquisition of software-intensive systems by focusing on acquisition programs, people, and production/sustainment and by institutionalizing continuous improvement.

This special report contains a briefing (slides and accompanying notes) on the results of one subtask of this effort conducted during FY06. The subtask called for three actions: (1) explore the (then) current processes and test results/metrics used to address system-of-systems integration and testing, (2) develop findings and recommendations for improvement based on this initial exploration, and (3) recommend future work to further improve the Army's system-of-systems integration and test practices.

The Army is in the lead in addressing the many challenges associated with system-of-systems integration and testing, paving the way for the rest of the U.S. Department of Defense (DoD). As a result, the information contained in this report is useful to other organizations facing similar challenges.

1 Background

The Army Strategic Software Improvement Program (ASSIP) is a long-term strategic partnership with the Assistant Secretary of the Army (Acquisition, Logistics, and Technology) (ASA(ALT), Program Executive Officers (PEOs), direct reporting Program Managers (PMs), Army Materiel Command Software Engineering Centers (SECs), and the Carnegie Mellon® Software Engineering Institute[2] (SEI). The ASSIP goal is to dramatically improve the acquisition of software-intensive systems by focusing on acquisition programs, people, and production/sustainment and by institutionalizing continuous improvement.

This special report contains information related to one subtask of this effort, conducted during FY06. Many challenges are associated with system-of-systems integration and testing. As a subject matter expert and neutral party, the SEI was engaged to

- explore the current processes and test results/metrics that are used to address system-of-systems integration and testing
- develop findings and recommendations for improvement based on this initial exploration
- recommend future work to further improve the Army's system-of-systems integration and test practices

The ultimate goal, not attained in this task, is for the Army to have processes and corresponding test results/metrics that are needed to address system-of-systems integration and testing, enabling senior leaders to make informed deployment decisions. This will involve Army acquisition organizations, program executive offices such as PEO C3T (Program Executive Office Command Control Communications Tactical), TRADOC (U.S. Army Training and Doctrine Command), CIO/G6 (Department of the Army, Office of the Army Chief Information Officer), G3 (Department of the Army, Office of the Deputy Chief of Staff for Operations & Plans), and G8 (Department of the Army, Resource Management). It also will involve the Army Test and Evaluation Command (ATEC) and the Army Central Technical Support Facility (CTSF). In short, there will be responsibilities and expectations from all the stakeholders in this arena.

In support of uncovering necessary information and background, the SEI interviewed key stakeholders and contributors to better understand

- what was there in the timeframe under review (beginning in April 2004) and at the time of the review (April-June, 2006)

- the challenges faced

- the solutions used (to the date of the review)

The Army is in the lead in addressing the many challenges associated with system-of-systems integration and testing, paving the way for the rest of the U.S Department of Defense (DoD). As a result, the information contained in this report is useful to other organizations facing similar challenges.

1.1   Acknowledgements

The task team was composed of Robert W. Ferguson (Software Engineering Measurement and Analysis Initiative), Margaret Glover (Acquisition Support Program), Patricia Oberndorf (Dynamic Systems) and Carol A. Sledge, Ph.D. (Dynamic Systems), Task Lead, from the SEI, with Mr. Mrunal Shah as the PEO C3T ASSIP liaison. The SEI task team conducted the task and formed the recommendations and associated materials contained in the original briefing (provided in July 2006) to the ASSIP Advisory Group.

The task team wishes to again thank all stakeholders interviewed for going out of their way to be helpful to us and providing

- their open and candid assessments, including background information

- access to materials, methods, processes, and results

- descriptions and demonstrations of current processes, and the like

1.2   Caveats

The SEI conducted its review with respect to a specific set of circumstances at a substantially later point in time (with respect to those circumstances): Our objective was to

- learn from what had occurred

- determine the reasons behind the decisions/events

- discover improvements made and lessons learned by the various stakeholders and organizations

Consequently, we prepared our briefing to

- use that information and knowledge to address the team's specific task

- provide recommendations going forward

This report consists essentially of the slides and accompanying notes from the latest version (Fall 2006) of the Army ASSIP System-of-Systems Test Metrics Task with some additional material. Reading the report is not equivalent to attending a briefing of these materials: the report is incomplete without the accompanying oral presentation and the opportunity to ask questions, seek clarifications, provide additional feedback, and so forth to prevent any misunderstandings or unintended conclusions.

As a result of this work, we found the challenges associated with system-of-systems integration and testing

- reach back to events and decisions (much) earlier in the acquisition, development and test life-cycles
- cross multiple organizations, numerous times
- are aggravated by the rapidity of technological change as well as the closer integration of existing assets and systems necessitated by the changing demands of the operational environment
- are magnified by all of the attendant elements and changes resulting from the transition from a system focus/basis to a system-of-systems focus

While specific organizations are mentioned in the recommendations, the recommendations are stated "looking forward, with a system-of systems-viewpoint" and should not be construed as a criticism of any organization.

Finally, this task was not done in isolation: other parallel tasks were/are in progress for the Army ASSIP Program.

2 Army ASSIP System-of-Systems Test Metrics Briefing

The original version of this briefing was given to the ASSIP Advisory Group (AAG) at their July 18, 2006 meeting. Subsequent versions of the briefing have reordered some of the recommendations and slides, added some information to some of the slides or notes, and, based on subsequent feedback (in September), added a ninth recommendation. Subsequent versions of the briefing were given to key stakeholders. The task team appreciates the feedback and additional information given at the July 18th and subsequent briefings. This version of the briefing does not contain all of the material from that feedback and additional information. In particular, it does not contain the results of the continued commitment to the improvement of processes, procedures, products, and the like. To be certain, many challenges still exist, and the SEI is involved in continuing efforts in this system-of-systems area.[3]

---

[3] Continuing and future efforts include ongoing work with the Army ASSIP Program, the work of the SEI's Integration of Software-Intensive Systems Initiative, related work in other SEI initiatives, and the SEI's planned System–of-Systems Test and Evaluation Consortium (SoSTEC) (http://www.sei.cmu.edu/programs/ds/sostec.html).
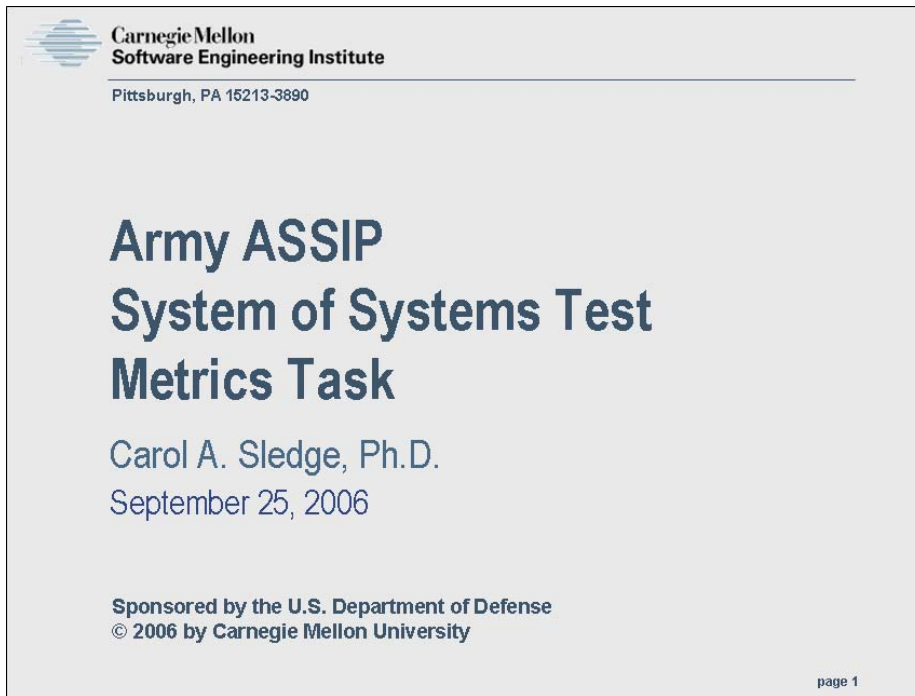
Figure 1: Briefing Title Page

The slides in this report represent the September 2006 update to the task report/briefing to the Army ASSIP Advisory Group (AAG) first given on July 18, 2006. The September 2006 update included some reordering of recommendations, some minor additions to some slides, and, based on feedback received from briefings to key parties subsequent to the AAG meting, the inclusion of a ninth recommendation.

Notes

- The information contained here-in is based on interviews and other materials obtained in the time period mid-April 2006 through mid-June 2006.
- The comments following each slide do not necessarily address all points shown on the slide.
- This special report also has some additional material (e.g. the three interoperability map examples in the Appendix) that was not part of the briefing.

Figure 2: Problem Statement

This was the problem that led to the funding of a brief task by the SEI to explore the current processes and test results/metrics.

Figure 3:   Statement of Initial Task

Many challenges are associated with system-of-systems integration and testing. As a subject matter expert and neutral party, the SEI has been engaged to explore the current processes and test results/metrics that are used to address system-of-systems integration and testing, develop findings and recommendations for improvement based on this initial exploration, and recommend future work to further improve the Army's system-of-systems integration and test practices. These recommendations could also include helping the Army to mature current good practices.

The ultimate goal, not attained in this task, is for the Army to have processes and corresponding test results/metrics that are needed to address system-of-systems integration and testing, enabling senior leaders to make informed deployment decisions. This will involve Army acquisition, PEO C3T, TRADOC, CIO/G6, and G8: there will be responsibilities and expectations from all the stakeholders in this arena.

In support of uncovering necessary information and background, the SEI will interview key stakeholders and contributors to better understand what is there today, the challenges they face and the solutions that they have used to date. SEI will coordinate with the Army Stakeholders (e.g., Army G3, G6, CTSF, C3T, ASA(ALT), G8, and TRADOC). Key to this exploration and understanding is the CTSF—both the integration and testing sides of the CTSF. The SEI will investigate what is delivered to the CTSF, what tasks, processes and services the CTSF performs, what entities the CTSF interacts with, what expectations the CTSF has, current good practices, and the like. Therefore, while most of the initial work will be accomplished via telephone interviews, a trip to Ft. Hood and the CTSF will be necessary.

Figure 4:   Approach and Time Period

The initial version of this annotated briefing/report was delivered to the ASSIP Advisory Group (AAG) on July 18, 2006.
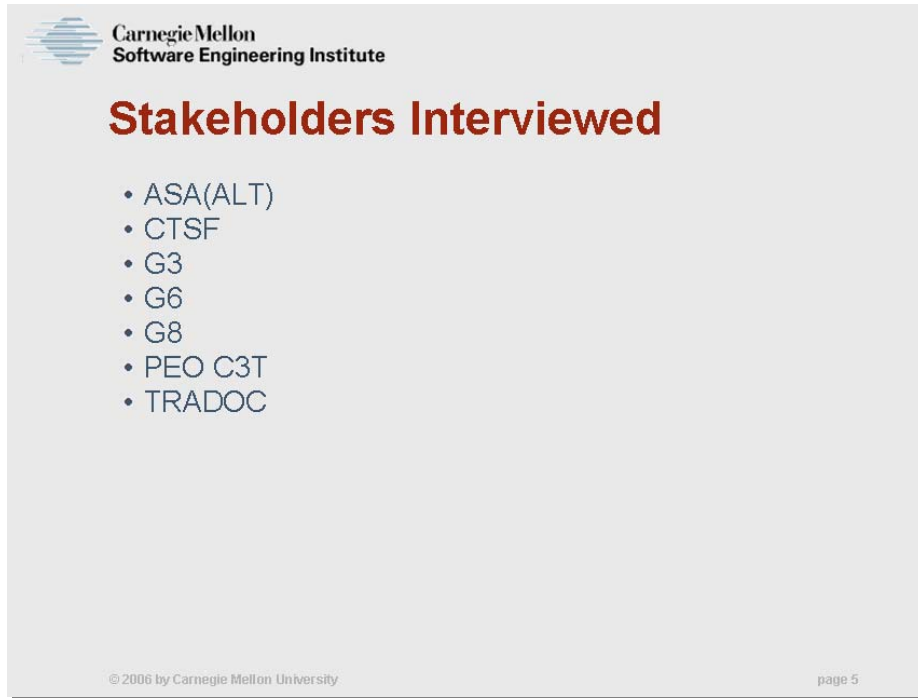
Figure 5:   Stakeholders Interviewed

Partial list of people interviewed

By phone, unless at CTSF

- COL Harry Greene, PEO C3T

- Janet Greenberg, PEO C3T

- Sylvia Sass TRADOC, TPIO, BC, Ft Leavenworth

- Terry Edwards, CIO G6

- G.J. "Skip" Stiles, HQDA CIO/G6

- Dr. James Linnehan, ASA(ALT)

- Celeste Kennamer, G3

- Beth Lynch, G8

At Whitfill Central Technical Support Facility (CTSF)/other personnel at Ft. Hood

- COL Charles McMaster, Director CTSF

- Ted Kostich, Technical Director

- Chuck Miller, Chief Systems Engineering & Integration

- Len Krals, Test Chief

- Robert Kidwell, Test (Chief, Certification Branch)

- Eugene Terwilliger, Test (Deputy and Interoperability Branch Chief)

- Don Kirby, Methodology Analysis Branch Chief

- Mike Curci, Configuration Management (CM) Chief

- Langston Carter, CM

- Joe Riggs, CM

- Justin Fielder, CM

- Mardy Turnbough, ORSA (Test) – Deputy Branch Chief, Methodology and Analysis

- Dr. Bill Stephens, Test, Information Assurance

- Phil Hallenbeck, Chief, Advanced Systems Engineering

- Eric Foster, Operations (Program Manager, Digital Systems Engineering)

- David Key, PEO C3T

- Stephanie Chenault

Post-July 18, 2006 ASSIP AAG briefing (due to scheduling constraints)

- Joan Smith, HQDA CIO-G6/ASA(ALT), Director, Interoperability Assessment & Certification [July 20, 2006]

Figure 6:   General Observations

These observations were made in the period mid-April through mid-June 2006.

- Focus is on supporting the warfighter versus longer term Army strategic goals (e.g., the Army's net-centric future).

- It is clear that there have been many lessons learned and incremental process improvements made. Actions on some of these lessons learned have become more apparent in the last 9 to 12 months.

- Processes, tools, and mindsets are still primarily system oriented rather than system-of-systems oriented. You cannot do meaningful test metrics for a system of systems until you have defined what it is that you want to achieve.

Figure 7:   First Recommendation

Establish an overarching executive with a clear system-of-systems vision, mandate, and funding. While there were many lessons learned and improvements made throughout the last two to three years, the acquisition, structure, funding, processes are still primarily system focused, not system-of-systems focused. This system-of-systems focus requires radical changes, some aspects of which have been started. What is fielded is a system of systems, but what are developed and tested are still primarily individual systems—with the collection of systems tested as a series of "data points" through primary paths.

Systems of systems are more than the sum of their individual systems: it may well be the case that in looking at the interaction between two systems, both systems are performing as originally specified, but there is a problem at the point of integration.

All stakeholders in the (life-cycle) process must buy in to the system-of-systems view and its implications: meeting of milestones, funding, evaluation, contracts, and the like. From a system-of-systems view, there is no overarching executive with the "teeth" to achieve the system-of-systems results. The Army is clearly in a transition state; but the future desired state of system of systems has not been clearly articulated, nor have the implications of this system-of-systems mentality (which span all areas).

Figure 8:   Second Recommendation

Define overarching system-of-systems capabilities. To achieve meaningful system-of-systems integration test metrics, there must first be an agreed-upon definition (and formal criteria) for what this collection of systems is expected to achieve. These overarching system-of-systems capabilities must be agreed to, documented, and built to—with integrated and coordinated planning and master schedules for all the systems participating at all points from conception until fielding (and beyond). Test criteria and tests can then be developed (and aggregated and prioritized) from these overarching system-of-systems capabilities. Successful completion of the tests associated with each of these system-of-systems capabilities would give an indication of the readiness of that particular capability. Capabilities will span systems. In aggregate, it would give an indication of the readiness of the system of systems to be fielded. While simple to state, this result is difficult to achieve.

Given the overarching system-of-systems capabilities, with this collective view and the tests designed for each of those capabilities, one should be able to raise up the level of reporting to those system-of-systems capabilities—beyond the particular glitch or problem that may be contributing to the failure of a particular test.

There must be a member of the Army team that is constantly looking at things from an overarching system-of-systems viewpoint, and in particular from an integration and testing point of view. Without this, it will be difficult to define the additional needs to support the system-of-systems integration and testing and the definition of appropriate system-of-systems integration metrics.

- **Warfighter should play a more active role in the development of System of Systems capabilities**. When the Initial Capabilities Document (ICD) is being developed by TRADOC,

and before validation by the Joint Requirement Oversight Council, the integration and test facility (CTSF), Program Executives, PMs or their representatives should also have input into the ICD as well as signatory approval of the document before it becomes the Capability Development Document (CDD). The integration and test facility representative should be one of the approvers of the CDD to better ensure that the system-of-systems capabilities and underlying requirements are testable and that the elicitation, analysis, validation and communication of the warfighter/Field User needs and expectations are present in the document.

A formal process should be in place that describes the required contents of both the ICD and the CDD, along with the approval requirements and those that have approval authority. TRADOC as well as the Field User shall also approve both documents to ensure correctness and completion. It is imperative that, from a system-of-systems point of view, the ICD and the CDD are correct and complete in their description of the Software Block that is being built in order for the warfighter to correctly complete his/her mission at every level that is necessary for a successful mission.

The operational environment and the factors that reflect overall customer and end-user expectations and satisfaction shall be defined. It is necessary that the operational system-of-systems concept of the Software Block be understood by developers, integrators, testers and ultimate end users. Members of the TRADOC System Manager (TSM), who represent the warfighter, are generally contractors, so the actual military warfighter is usually not represented in the current generation of test threads/test requirements.

Although it is impossible to understand and completely capture all possible end-user requirements until a Software Block enters the field, the end-user is the primary component for the success of this mission. There appears to be a disconnect between the requirements gatherers and the end users. It is felt that by the time the requirements are gathered and implemented into a Software Block, those requirements are antiquated. TRADOC is seen as the primary group that gathers the requirements for the Software Block. Our recommendation is that the actual warfighter play a more active role in the determination of the requirements, specifically in writing the ICD. The Digital Systems Engineers would be useful complement to the actual warfighters as contributors to this process.

Define a formal test plan that represents the system-of-systems testing. A Formal Test Plan (FTP) is one of the documents that would greatly benefit the system-of-systems formal test. The FTP will reflect the requirements set forth in the (System of Systems) Capability Development Document (CDD). Various integration and test facility representatives will analyze each capability/requirement to understand how the capability/requirement will be tested, as well as the resources needed to test each requirement.

The testing schedule, that would be included in the test plan, would ensure that the testing is not just done as it is now, to a "drop dead date." The only quality measurement that indicates the end of the test phase is the definition of "good enough." This qualifier needs to be defined, measured, and documented in the test plan. The test plan should include the

- test set up

- test cases and their ties to the requirements

- test values that are expected

- planned time for each test case with an overall work breakdown structure (WBS) for all test cases

The test plan should also detail how defects will be categorized, documented, assigned, and communicated.

Figure 9:   Third Recommendation

Preliminary system-of-systems integration should occur before delivery to the formal integration and test facility. Relating back to the defined system-of-systems capabilities and any subordinate, supporting capabilities, the individual systems must be designed to and tested against those required capabilities. Just as the capabilities are overarching for a system of systems, there should be a defined and supported simulator/stimulator that can be used by a particular system to help test conformance to the system-of-systems capabilities. This should substantially reduce the amount of test-fix-test time at the integration and test facility. This assumes that there is strict system-of-systems change management/change control for the systems that participate in the system of systems, so that there are no "surprises" regarding changes to capabilities/implementation of same.

- **Interim: Formal system–of-systems integration test phase at the integration and test facility (CTSF).** It appeared as if there was no Integration Test Phase that is defined for the system-of-systems CTSF testing. Integration Test does not occur until the various systems arrive at CTSF and start to be worked together in a test-fix-test environment. It seems that Integration Test is expected to have happened before it gets to CTSF. It appeared that CTSF was the only place that integration test could occur because of the large and complex task of the integration of all these systems that can and do come to CTSF for system-of-systems testing. It is therefore recommended that an Integration Phase be formally defined, planned, and implemented as a separate phase at CTSF before system- of-systems test gets underway. Measurements shall be taken and reported for schedule (such as time, test cases, integration tasks, and personnel) as well as defect metrics, along with a Configuration Management function of the software and hardware undergoing Integration Test as it is being built into a system of systems.

Figure 10: Fourth Recommendation

Use an integrated defect tracking system. Each problem or defect reported in a system of systems may affect other component systems. The fix to a problem may be engineered in a component system other than the component system that caused the problem. The important theme is that all participants need access to information about the problems and defects no matter what the origin or source of the problem.

Ancillary needs include: formalized defect process, official point of contact (POC) in each PEO/PMO, and reporting about both process and defects. Infrastructure support for defect management such as in a system-of-systems defect tracking tool is also needed. The Test Incident Reports (TIR) database exists and is used, but it was felt that it only represents system defects, not system-of-systems defects. The TIRs give only a low level view of what is being captured as defects. Many of the TIRs are not functional problems with the threads, but problems with the database, and an I/O handshake, a subscribe problem, and the like.

System-of-systems defects that are found by testing the functional threads should be detected to represent the threads at a higher level of the warfighter functionality/capability. They should be tracked and searched on by Date, Test Case #, Requirement #/Thread #, Priority, Found by, Assigned Too, Status (open, closed, not repeatable, not valid) and Nature of the Problem.

In defining the integrating defect tracking system, keep in mind the immediate needs and inputs and the longer-term system-of-systems capabilities (versus individual system requirements) implications.

Figure 11: Fifth Recommendation

Establish and enforce (system of systems) minimal/threshold requirements/entry criteria for each system being delivered to the formal integration and test facility. To reduce the impact on other systems participating in the system of systems and to facilitate the testing of the system-of-systems capabilities, these must be established and enforced. This also will require coordinating schedules and "demonstration" or delivery of information prior to the formal delivery of the system to the formal integration and test facility. This also implies that the overarching system-of-systems master schedule has given the particular system sufficient time to do the development, unit test, simulated (or real) integration tests (with respect to capabilities and interfaces), prior to the date for system delivery to the formal integration and test facility. Failure to meet these agreed to entry requirements will result in the system not being accepted for testing.

- **System-of-systems testing requires stable systems working from a known baseline.** This is a long-standing principle for testing large systems. The development team must stabilize the system for the period of testing. A consolidated package of defects is needed for all but those defects that prevent continued testing. During Software Block 1, many systems were not stabilized. Allowing too many changes to hit the test team at unplanned intervals can compromise the integrity of the entire system. It also causes the test team to spend far too much time (opportunity cost) on installation and regression tests. Thus testing throughput is again diminished and there are fewer opportunities to appropriately identify and diagnose new defects.

  Likewise, changes/fixes to systems "late to the party" adversely affect systems that met schedule and stability. This means of course that the systems composing the system of systems have planned for, are funded for, and have agreed to a system-of-systems master schedule that will enable them to be stable on delivery to the integration and test facility.

Figure 12: Sixth Recommendation

(There are not any notes associated with this slide.)

Figure 13: Seventh Recommendation

Develop both system-of-systems test progress metrics and quality/"goodness" metrics. Progress metrics: test progress during the test process itself – are things moving forward? Quality/"goodness" metrics are related to progress, but they are separate—system-of-systems quality/"goodness" metrics. Neither of these metrics is an evaluation of CTSF.

• **Integration and test facility provides regular information about testing progress.** The system-of-systems overarching executive, in addition to the PEOs and PMs, must see evidence of testing progress in order to make projections about system readiness to their stakeholders. This reporting would be independent of defect tracking. Completion of all tests and an appropriate quality score are necessary to complete certification.

Regular metrics reporting might include some or all of the following:

− test case schedule updated at least weekly
− test cases attempted
− test cases completed (both pass and fail)
− testing throughput (weekly performance)
− test cases to go for certification

(The relationship of mission threads to test threads to test cases is as follows: test threads represent ways the mission might happen and test cases are examples where the configuration might be something different. The "hierarchy" is mission thread – test thread – test case.)

- **System-of-systems testing should utilize both small and large test threads.** Current CTSF practice is to utilize very large threads almost exclusively. Since both defects and testing progress are essential to the overarching system-of-systems executive, program executive and program management, it is important to demonstrate that tests are completed whether pass or fail.

Software Block 1 included about 150 mission threads. The number of mission threads may be appropriate. That question needs a separate discussion, as does the actual selection of mission threads. It is not possible for the SEI to make a determined judgment of the number and quality of the tests associated with these mission threads without an in-depth analysis that was far beyond the scope of the current investigation. However, the relatively small number of test cases and the complexity of each one are concerns because the measurement of testing throughput can be questioned where there is so little data to examine.

Recommendation 8

Define initial indicators/metrics
- agree on a System of Systems defect taxonomy
- if continued use of "stoplight" – red/yellow/green
  - include definitions of meaning of red/yellow/green
  - effect of defect color on color of overall test thread
  - effect of colors of test threads on color of overall mission thread

© 2006 by Carnegie Mellon University   page 14

Recommendation 8 (continued)

Define initial indicators/metrics
- example potential indicators[1]
  - percent of thread tested and green
  - percent of threads that are complete and green
  - percent of critical mission threads complete
  - percent of total mission threads complete
  - number of tests executed [progress trends]
  - number of tests that remain to be executed [progress trends]
  - test throughput
  - elapsed time between identification of defect and correction
  - test incident report aging
  - number of components changed rate [testing input]

[1] *Applications of the Indicator Template for Measurement and Analysis*

© 2006 by Carnegie Mellon University   page 15

Figure 14: Eighth Recommendation

Note

The potential indicators in recommendation eight are simply examples: the list is not meant to be inclusive, to include "mandatory" indicators, or indeed to include the most obvious indicators. Also see the comments for recommendation seven.

Useful information in this area is contained in this technical note:

Goethert, Wolfhart & Siviy, Jeannine. *Applications of the Indicator Template for Measurement and Analysis* (CMU/SEI-2004-TN-024, ADA443479). Pittsburgh, PA : Software Engineering Institute, Carnegie Mellon University, 2004).

The abstract for this report is as follows :

> *Organizations often do not achieve the potential benefits of a sound measurement program due to the inconsistent construction and interpretation of indicators derived from measurement data. This technical note presents guidance for adapting and completing an indicator template—a tool the Software Engineering Institute has developed to precisely describe an indicator—including its construction, correct interpretation, and how it can be utilized to direct data collection and presentation and measurement and analysis processes. An indicator template can help an organization to define indicators, or graphical representations of measurement data, which describe the who, what, where, when, why, and how for analyzing and collecting measures. This technical note defines each field of the indicator template, provides example inputs, and shows how the template may be used in the context of a process improvement effort that uses the Capability Maturity Model® Integration framework and/or Goal-Driven Software Measurement.*

The above abstract and link to a portable document format (pdf) version of the actual document is available at http://www.sei.cmu.edu/publications/documents/04.reports/04tn024.html. This document is available for download at http://www.sei.cmu.edu/pub/documents/04.reports/pdf/04tn024.pdf.

Figure 15: Recommendations for Shorter-Term Improvements

Interoperability maps are a way to understand the relationships and dependencies among systems and program offices.

The Appendix of this report contains examples of the three interoperability map types: context, node-centric, and arc-centric. The constituent nodes in the maps can be facilities, organizations, systems, or components. Here are some greatly simplified examples of how interoperability maps can be useful.

- By using an appropriate context interoperability map, one can look at cascading effects.

- In an associated node-centric interoperability map, the near neighbors become surrogates for any other nodes the center node doesn't "see."

- An associated arc-centric interoperability map expands the exact nature of the agreements/conditions that need to be true for two nodes to have interoperability.

More information on interoperability maps can be obtained from this technical note:

Brownsword, Lisa; Fisher, David; Morris, Ed; Smith, James; & Kirwan, Patrick. *System-of-Systems Navigator: An Approach for Managing System-of-Systems Interoperability* (CMU/SEI-2006-TN-019). Pittsburgh, PA : Software Engineering Institute, Carnegie Mellon University, 2006.

The abstract for this report is as follows:

> *We have crossed a threshold where most of our large software systems can no longer be constructed as monoliths specified by a single, focused, and unified team; imple-*

mented as a unit; and tested to be within known performance limits. They are now constructed as groups of interoperating systems (as systems of systems) developed by different but sometimes related teams and made to interoperate through various forms of interfaces. Unfortunately, while we can easily conceive these large systems of systems, we have trouble building them. Software engineering practices have not kept pace, and the problem will only get worse as the community begins to build Internet-scale systems of systems like the Global Information Grid.

This technical note introduces the System-of-Systems Navigator (SoS Navigator), the collection and codification of essential practices for building large-scale systems of systems. These practices have been identified through the work of the Integration of Software-Intensive Systems Initiative at the Carnegie Mellon Software Engineering Institute. SoS Navigator provides tools and techniques to characterize organizational, technical, and operational enablers and barriers to success in a system of systems; identify improvement strategies; and pilot and institutionalize these strategies.

The abstract and link to a pdf is available at
http://www.sei.cmu.edu/publications/documents/06.reports/06tn019.html.The technical note is available for download at http://www.sei.cmu.edu/pub/documents/06.reports/pdf/06tn019.pdf.

Figure 16: Recommendations for Longer-Term Improvements

Figure 17: Ninth Recommendation

(There are not any notes associated with these slides.)

3 Summary

The Army Strategic Software Improvement Program (ASSIP) is a long-term strategic partnership with the Assistant Secretary of the Army (Acquisition, Logistics, and Technology) (ASA(ALT), Program Executive Officers (PEOs), direct reporting Program Managers (PMs), Army Materiel Command Software Engineering Centers (SECs), and the Carnegie Mellon Software Engineering Institute (SEI). The ASSIP goal is to dramatically improve the acquisition of software-intensive systems by focusing on acquisition programs, people, and production/sustainment and by institutionalizing continuous improvement.

This special report contains information related to one subtask of this effort, conducted during FY06. Many challenges are associated with system-of-systems integration and testing. As a subject matter expert and neutral party, the SEI was engaged to

* explore the current processes and test results/metrics that are used to address system-of-systems integration and testing
* develop findings and recommendations for improvement based on this initial exploration
* recommend future work to further improve the Army's system-of-systems integration and test practices

While specific organizations are mentioned in the recommendations, the recommendations are stated "looking forward, with a system-of-systems viewpoint" and should not be construed as a criticism of any organization.

The Army is in the lead in addressing the many challenges associated with system-of-systems integration and testing, paving the way for the rest of the U.S Department of Defense (DoD). As a result, the information contained in this report is useful to other organizations facing similar challenges.

Appendix   Examples of Interoperability Maps

The following material describes interoperability maps. It is taken verbatim from the technical note *System-of-Systems Navigator: An Approach for Managing System-of-Systems Interoperability* (CMU/SEI-2006-TN-019) referenced on page 24. The numbering of figures and footnotes has been adjusted from the original for the layout of this report.[4]

Interoperability Maps

Interoperability maps characterize the relationships in a system of systems from three perspectives: (1) the global system-of-systems entity, (2) individual constituents (represented as nodes in the graph), and (3) individual agreements (represented as arcs between a pair of nodes).

These maps permit the capture of information about how constituents actually influence one another in a system of systems. That is, they portray the reality of how things work—not how they are supposed to work—and represent actual understandings, intents, and expectations of constituents, as opposed to what is stated in acquisition and design artifacts.

There are three forms of interoperability maps currently in the SoS Framework[5] element.

1. **Context Interoperability Map** depicts high-level, system-of-systems-wide information about contractual, funding, requirements, hardware, oversight, and build/integrate influence relationships from the global system-of-systems entity perspective.

2. **Node-centric Interoperability Map** portrays information about contractual, funding, requirements, hardware, oversight, and build/integrate influence relationships from the perspective of individual constituents.

3. **Arc-centric Interoperability Map** represents information about needs, offers, expectations, intentions, and negotiated agreements between the constituents involved in the influence relationship.

Context Interoperability Map

The nodes represented in the Context Interoperability Map are

---

4   More information on interoperability maps and the System-of-Systems Navigator approach (called the SoS Navigator) can be obtained from that document.
5   The SoS Framework is a "diverse set of core paradigms and principles along with processes and techniques" developed by the SEI Integration of Software-Intensive Systems Initiative (http://www.sei.cmu.edu/pub/documents/06.reports/pdf/06tn019.pdf).

- systems

- major management entities (e.g., contractors, program offices, or agencies)

- funding organizations (e.g., appropriations committees)

- oversight organizations (e.g., regulatory boards or standards bodies)

- contractual organizations

As the Context Interoperability Map presented in Figure 18 illustrates, arcs connect nodes that have an influence relationship.[6] These influence relationships can be highly complex, encompassing multiple dimensions of schedule, contracting, and performance. The Context Interoperability Map conveys a general "lay of the land" and may also provide insight into possible areas of the system of systems that would be good candidates for further exploration.
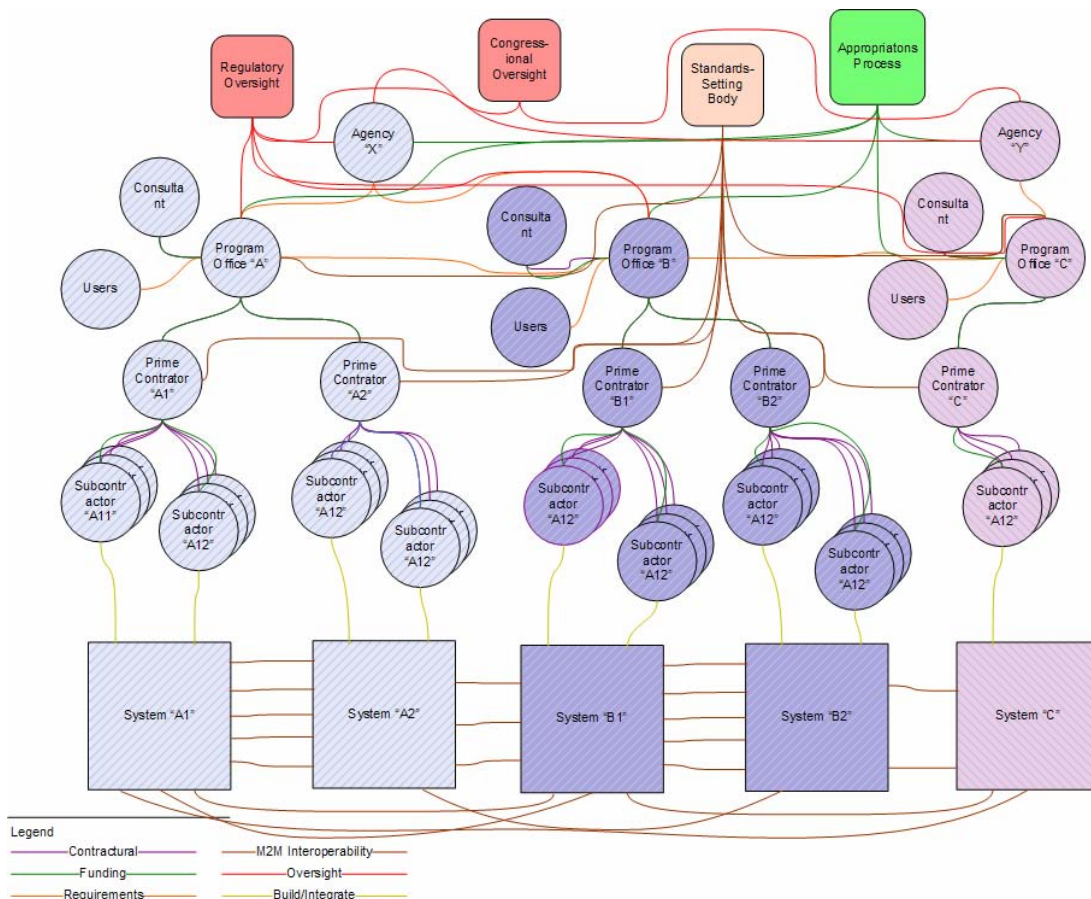

Figure 18: Context Interoperability Map

The Context Interoperability Map allows the SoS Navigator team to capture the broad influences on the system of systems. In effect, this graph represents the viewpoint of the system-of-systems global entity responsible for the overall system of systems. It identifies and documents many in-

---

6   The interoperability maps shown in this section are conceptual models designed to illustrate the kinds of maps produced in the SoS Framework element. They do not reflect an actual system or system effort.

dividual constituents that participate in the systems-of-systems effort. However, it does not attempt to identify all of the influences that impinge on individual nodes; that is the function of the Node-centric Interoperability Map.

Node-Centric Interoperability Map

From the standpoint of a constituent, the Node-centric Interoperability Map (shown in Figure 19) documents the influences in a system of systems.
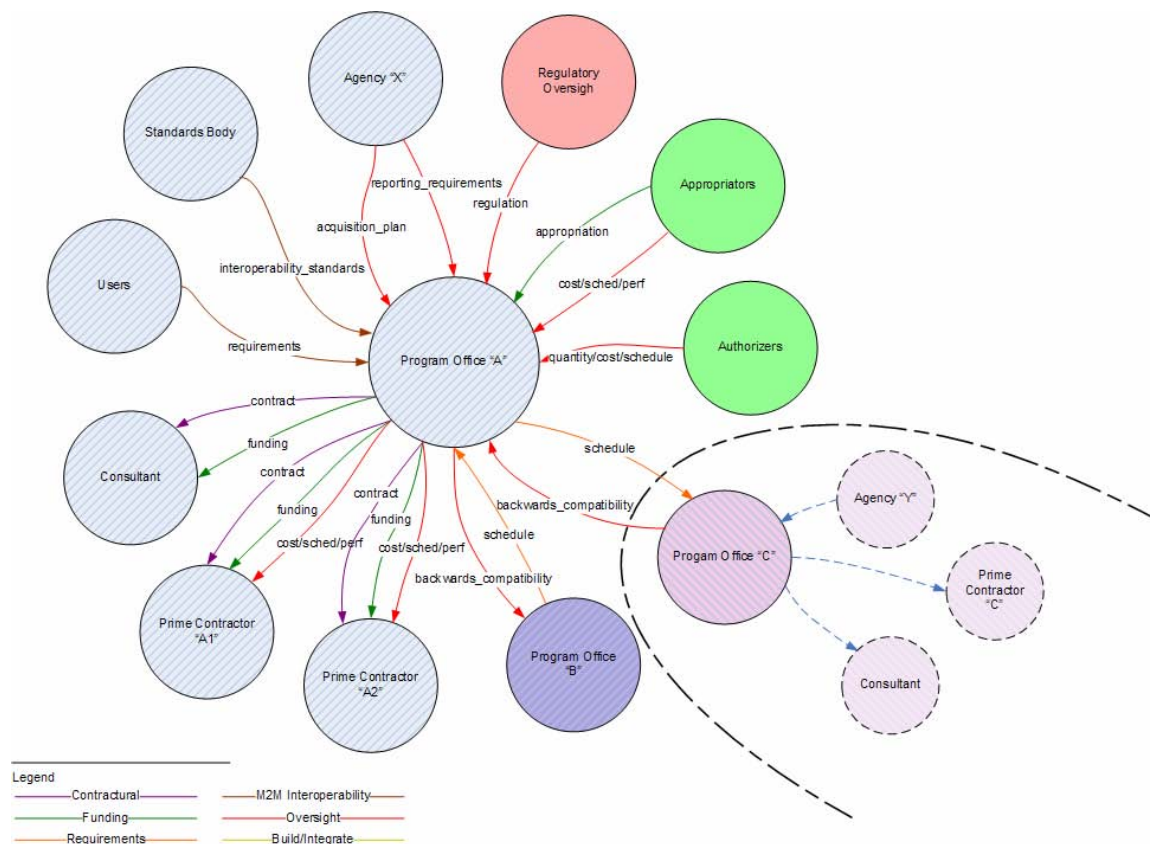

Figure 19: Node-centric Interoperability Map

Node-centric Interoperability Maps are specialized to the perspective of a single program management office, contractor, or other type of constituent. They reveal what is "visible" to the constituent. An important aspect is that a constituent represents the relevant interests of "downstream" constituents to an "upstream" constituent. For example, in Figure 19, Program Office "C" would represent any schedule constraints that it has with any downstream constituents (Agency "Y" and Prime Contractor "C") in its schedule relationship with Program Office "A." This notion of pass-through or transitive influences allows the influence relationships affecting a particular constituent to be understood without requiring that constituent to have insight into the entire system of systems.

Figure 19 shows how influence relationships can be fairly complex:

- The direction of an arc represents the primary direction of influence.

- The destination node of each arc (i.e., the "upstream" constituent) has a need that represents the claimed minimal set of critical expectations from the source node of the arc (possibly as function of schedule, value, or quality).

- The arc source node has an offer that represents the broadest set of relevant things that it can feasibly provide to the destination node.

- Each arc has an associated agreement that may be in part implicit, informal, or tacit.
    - Agreements derive from negotiation—often informal—of needs and offers.
    - Agreements may be vague initially and then refined as detail is needed and understood.
    - In combination with the context in which the neighbors operate and the trust they place in their partners, agreements determine the intents and expectations along each arc.

Node-centric Interoperability Maps provide a mechanism to establish consistency between what one constituent believes to be important interrelationships (as reflected in the Context Interoperability Map) and what other constituents believe to be important.

In addition to providing sufficient detail to support the analysis of inconsistencies and conflicts, the Node-centric Interoperability Maps identify relationships to organizations outside of the purview of a global system-of-systems entity. For example, Figure 19 represents relationships between Program Office "A" and several constituents not normally under the purview of most global system-of-systems entities (i.e., appropriators, authorizers, and regulatory oversight bodies). Notice that these constituents can have a significant impact on a system of systems but often are not considered.

Arc-Centric Interoperability Map

Arc-centric Interoperability Maps express and make explicit the (often implicit) assumptions that go into an influence relationship. They can be used in situations where influence relationships are particularly complex, critical, or easily misunderstood. In an Arc-centric Interoperability Map, as Figure 20 demonstrates, the needs of the requesting constituent are expressed as a set of minimum critical needs (MCNs)—the absolute minimum that is truly necessary to satisfy the requestor's constraints. The response from the offering constituent is expressed as a set of broadest feasible offers (BFOs)—the "most generous" response it can provide that does not violate its constraints.
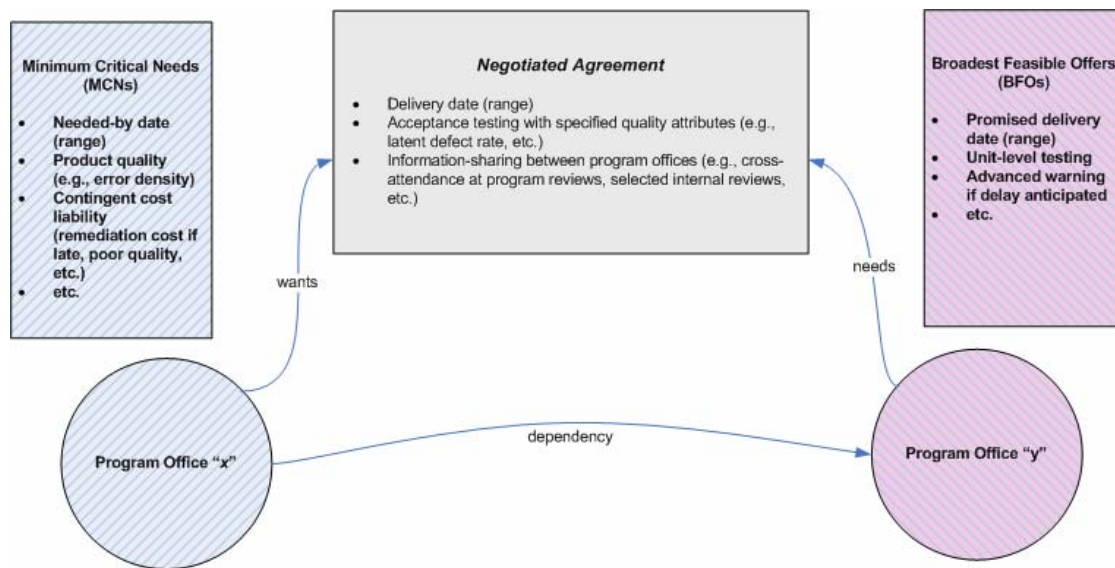
Figure 20: Arc-centric Interoperability Map

Where there is an overlap between the MCNs and BFOs, an agreement is possible; where there is no overlap, no feasible match between the requestor's needs and the offering constituent's capabilities exists. In short, no overlap—even after negotiating (i.e., exploring whether restating needs and offers can possibly result in an overlap)—indicates that no agreement is possible. The focus of Arc-centric Interoperability Maps on MCNs and BFOs is important, because those assumptions represent the end points of a range within which a negotiated agreement is possible. Interestingly, these end points are often not the same as the negotiated agreement, since the agreement often represents a more optimistic view of events.

REPORT DOCUMENTATION PAGE

1. AGENCY USE ONLY	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
(Leave Blank)	November 2006	Final

4. TITLE AND SUBTITLE	5. FUNDING NUMBERS
Army ASSIP System of Systems Test Metrics Task	FA8721-05-C-0003

6. AUTHOR(S)

Carol A. Sledge, PhD

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
Software Engineering Institute Carnegie Mellon University Pittsburgh, PA 15213	CMU/SEI-2006-SR-011

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY REPORT NUMBER
HQ ESC/XPK 5 Eglin Street Hanscom AFB, MA 01731-2116	

11. SUPPLEMENTARY NOTES

12A DISTRIBUTION/AVAILABILITY STATEMENT	12B DISTRIBUTION CODE
Unclassified/Unlimited, DTIC, NTIS	

13. ABSTRACT (MAXIMUM 200 WORDS)

The Army Strategic Software Improvement Program goal is to dramatically improve the acquisition of software-intensive systems by focusing on acquisition programs, people, and production/sustainment and by institutionalizing continuous improvement.

This special report contains a briefing (slides and accompanying notes) on the results of one subtask of this effort conducted during FY06. The subtask called for three actions: (1) explore the (then) current processes and test results/metrics used to address system-of-systems integration and testing, (2) develop findings and recommendations for improvement based on this initial exploration, and (3) recommend future work to further improve the Army's system-of-systems integration and test practices.

The Army is in the lead in addressing the many challenges associated with system-of-systems integration and testing, paving the way for the rest of the U.S. Department of Defense (DoD). As a result, the information contained in this report is useful to other organizations facing similar challenges.

14. SUBJECT TERMS	15. NUMBER OF PAGES
System of systems, test, evaluation, ASSIP, acquisition	44

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL